

Moving Object Detection in Video Using Saliency Map and Subspace Learning

Yanwei Pang^{*a}, Li Ye^a, Xuelong Li^b, and Jing Pan^{a,c}

^aSchool of Electronic Information Engineering, Tianjin University, Tianjin 300072, China

^bInstitute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China

^cSchool of Electronic Engineering, Tianjin University, Tianjin 300222, China

Abstract

Moving object detection is a key to intelligent video analysis. On the one hand, what moves is not only interesting objects but also noise and cluttered background. On the other hand, moving objects without rich texture are prone not to be detected. So there are undesirable false alarms and missed alarms in many algorithms of moving object detection. To reduce the false alarms and missed alarms, in this paper, we propose to incorporate a saliency map into an incremental subspace analysis framework where the saliency map makes estimated background has less chance than foreground (i.e., moving objects) to contain salient objects. The proposed objective function systematically takes account into the properties of sparsity, low-rank, connectivity, and saliency. An alternative minimization algorithm is proposed to seek the optimal solutions. Experimental results on the Perception Test Images Sequences demonstrate that the proposed method is effective in reducing false alarms and missed alarms.

1. Introduction

Object detection is the basis of intelligent video analysis. Generally, object recognition, action and behavior recognition, and tracking rely on the detected objects. In a sequence of images, there are both moving and static objects. In this paper, the focus is on detecting moving objects in a video. Moving object detection is related to but also different from class-specific object detection and general salient object detection. Pedestrian detection, face detection, and hand detection are instances of class-specific object detection. The task of moving object detection is to detect semantically meaningful moving objects. Pre-defined classes of moving objects should be detected by a moving object detection algorithm. Moreover, other se-

mantically meaningful objects should also be detected even though their classes are not pre-defined. Examples of meaningless moving objects include water ripples, waving trees (leaves), shadows, noisy data, and the one caused by variations of illumination. However, the moving object detection algorithm relying merely on motion information is prone to incorrectly classify such meaningless moving objects as meaningful ones. The corresponding error is called false alarms. But a salient object detection algorithm tends to correctly discard the meaningless objects. Hence, in this paper, we propose to incorporate the output (i.e., saliency map) of a salient object algorithm into a subspace analysis based objective function so that the problem of false alarms can be alleviated. It is noted that our method is also capable of alleviating the problem of missed alarms. Existing moving object detection algorithms tend to classify flat regions (i.e., textureless regions) inside an object and moving regions with similar appearance (texture) to background as static background and thus such regions may be missed. State-of-the-art salient object detection algorithm can output large value of saliency map at such regions. Utilizing the saliency map, our method has ability to classify such regions as foreground. In summary, we present an objective function that unifies subspace analysis of background and saliency map. The objective function consists of four terms: saliency map, sparsity, connectivity, and low-rank. An alternative minimization algorithm is proposed to find the optimal solution. The significant advantage compared to previous subspace based approaches is that saliency map is used to guide the result to have less false and missed alarms. The proposed method is named MODSAM. It is natural that ideal saliency map (e.g., the bottom of Fig. 1(a) and Fig. 1(b)) is desirable for the proposed method. However, even relatively unsatisfying saliency map (e.g., the bottom of Fig. 1(c) and Fig. 1(d)) can also play a positive role in the proposed MODSAM method. Of course, completely bad saliency map has a negative influence on moving object detection. Fortunately, great progress of salient object

^{*}pyw@tju.edu.cn

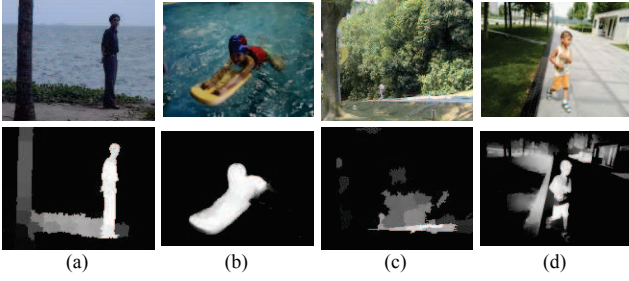


Figure 1: Images (top) and their saliency maps (bottom).

detection has been achieved [27, 14] and their fruits can be borrowed for moving object detection.

Several methods were developed to employ a salient object detection algorithm for improving the performance of moving object detection [10], [29], [33]. Despite the initial success, their performance cannot arrive at the level of state-of-the-art low-rank based and subspace based methods [34], [35], [11], [30], [8], [23], [9].

The rest of the paper is organized as follows. We review related work in Section II. The proposed method is given in Section III. Experimental results are provided in Section IV. We then conclude in Section V.

2. Related work

Moving object detection can be implemented by different manners: detecting followed by tracking [19, 6, 13], subtracting frames [18, 28], modeling background by density function, modeling background by subspace, modeling background by low-rank matrix. The last two manners dominate the state-of-the-art methods and are closely related to our work. Note that moving object detection methods can also be divided into incremental methods and batch methods. Our method belongs to incremental one.

Subtracting frames This kind of methods detects moving objects based on the differences between adjacent frames [18], [28]. But these methods were proved not robust against illumination variations, changing background, camera motion, and noise.

Modeling background by density function This strategy assumes that the background is stationary and can be modeled by Gaussian, Mixture of Gaussians, or Dirichlet Process Mixture Models [12], [15], [9]. The foreground (moving objects) can then be obtained by subtracting the current frame with the background model.

Modeling background by subspace Instead of using a density function, subspace based method models the background as a linear combination the bases of a subspace [11], [30], [8], [32], [20]. Because the subspace can be updated in an incremental (online) manner, its efficiency is very high. This kind of subspace based algorithms needs to impose constraints on the foreground in order to obtain

valid solutions. Foreground sparsity is one of the widely used constraints which implies that the area of moving objects is small relative to the background. Principal Component Pursuit (PCP) [3] is an important pioneer work which adopts norm for measuring the foreground sparsity. It is the constraint of foreground sparsity that makes PCP suitable for foreground-background separation. Without this constraint, traditional robust subspace methods can only deal with noise and outliers [7], [25], [22], [26]. The method [31] improves PCP by taking into account the foreground connectivity (i.e., foreground structure). RFDSA takes into account smoothness and arbitrariness constraints [8].

But PCP [3], RFDSA [8], and the method [31] are batch algorithms. Its detection speed cannot arrive at real-time level. Therefore, incremental (online) subspace methods are crucial for real-time detection [2]. He et al. [11] proposed an online subspace tracking algorithm called GRASTA (Grassmannian Robust Adaptive Subspace Tracking Algorithm). Similar to PCP, GRASTA also explores norm for imposing sparsity on foreground. But the GRASTA algorithm does not utilize any connectivity (a.k.a., smoothness) property of foreground. The GOSUS (Grassmannian Online Subspace Updates with Structured-sparsity) algorithm [30] imposes a connectivity constraint on the objective function by grouping the pixels with a superpixel method and encouraging sparsity of the groups. Because of the large computational cost of the superpixel algorithm [1], GOSUS is not as efficient as GRASTA.

Modeling background by low-rank matrix Low-rank modeling is effective in video representation [34]. A sequence of vectorized images is represented as a matrix and the matrix is approximated by the sum of matrices of vectorized foreground, background, and noise [35]. It is reasonable to assume that the background matrix is low-rank. DECOLOR (DEtecting COntiguous Outliers in the LOW-rank Representation) [35] is considered as one of the most successful low-rank based algorithms. In DECOLOR, both foreground sparsity and contiguity (connectivity) are taken into account. It can be interpreted as ℓ_1 -penalty regularized RPCA [35]. But the matrix computation can be started only if all of the predefined number of successive images is available. Obviously, such a batch method is not suitable for real-time video analysis due to its low efficiency. ISC [21] and COROLA [23] are incremental versions of DECOLOR. ISC and COROLA transforms low-rank method to subspace one.

The low-rank methods and subspace methods impose sparsity and connectivity (a.k.a., smoothness) on foreground and impose low-rank or principal components on background. In addition to such properties, in this paper we propose to impose saliency map on background and foreground meanwhile.

3. Proposed method

The proposed method belongs to incremental subspace based moving object detection method. Our main contribution lies in employing a saliency map to form a new objective function, resulting in fewer false and missed alarms.

3.1. Input and Output

The input of the Algorithm is a sequence of frames (images). Denote $\mathbf{o} \in \mathbb{R}^{N \times 1}$ the current image and denote o_i the i -th pixel of \mathbf{o} . There are N pixels in an image. The goal is to find the locations of the moving objects (i.e., foreground) in the current image \mathbf{o} . The foreground locations are represented by a foreground-indicator vector $\mathbf{f} \in \{0, 1\}^N$. The i -element f_i of \mathbf{f} equals to either zero or one:

$$f_i = \begin{cases} 0 & \text{if pixel } i \text{ is classified as background,} \\ 1 & \text{if pixel } i \text{ is classified as foreground.} \end{cases} \quad (1)$$

The foreground-indicator vector \mathbf{f} is obtained by binarizing background vector $\mathbf{b} \in \mathbb{R}^{N \times 1}$ with a threshold t :

$$f_i = \begin{cases} 0 & \text{if } b_i \geq t, \\ 1 & \text{if } b_i < t, \end{cases} \quad (2)$$

where b_i is the i -element of \mathbf{b} . The possibility of pixel i being background increases with the value of b_i and the possibility of pixel i being foreground decreases with the increasing value of b_i .

3.2. Problem Formulation

As stated above (i.e., Eq. (1) and Eq. (2)), the foreground-indicator vector can be obtained by binarizing background vector \mathbf{b} . The problem is how to compute \mathbf{b} once a frame \mathbf{o} is given. In this paper we formulate the problem of computing \mathbf{b} as the following minimization problem:

$$\min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta (1 - b_i) - \alpha b_i (1 - s_i) \right] + \lambda \|\mathbf{D}\mathbf{b}\|_1, \quad (3)$$

where \mathbf{U}_i stands for the i -th row of \mathbf{U} . In Eq. (3), $s_i \in [0, 1]$ is the i -th element of the vector $\mathbf{s} \in \mathbb{R}^{N \times 1}$ of a saliency map obtained by some salient object detection algorithm such as [36]. The value of s_i reflects the confidence that the pixel i belongs to a salient object. The term $-b_i(1 - s_i)$ is called saliency map term.

Minimizing the term $-b_i(1 - s_i)$ makes the estimated background \mathbf{b} has less chance than foreground to contain salient objects. Moving objects such as pedestrian, car, dog are indeed salient objects in a video. Therefore, the proposed method is capable of making estimated foreground

to have high-level semantic objects and fewer false alarms. The saliency map term $-b_i(1 - s_i)$ is the main novelty of the paper. α is the weight of the saliency map term. In Table 2, an empirical method for setting α is given.

In addition to $-b_i(1 - s_i)$, there are three terms: $b_i(\mathbf{U}_i \mathbf{v} - o_i)^2$, $(1 - s_i)$, and $\|\mathbf{D}\mathbf{b}\|_1$ which are to be described as follows. The weights for $(1 - s_i)$ and $\|\mathbf{D}\mathbf{b}\|_1$ are respectively β and λ whose values can be assigned according to Table 2.

The term $b_i(\mathbf{U}_i \mathbf{v} - o_i)^2$ is called background reconstruction term [11], [30]. In this term, $\mathbf{U} \in \mathbb{R}^{N \times m}$ is a subspace matrix and m is the number of columns of \mathbf{U} . The vector $\mathbf{v} \in \mathbb{R}^{m \times 1}$ is called coefficient vector. Minimizing the term $b_i(\mathbf{U}_i \mathbf{v} - o_i)^2$ makes the reconstructed background approaching the input frame \mathbf{o} as possible as it can.

The term $(1 - s_i)$ is called foreground sparsity term. Minimizing $\beta(1 - s_i)$ makes the estimated foreground much sparser than background.

The term $\|\mathbf{D}\mathbf{b}\|_1$ is called connectivity term [30], [8]. The matrix $\mathbf{D} \in \mathbb{R}^{2N \times N}$ is a difference matrix [30], [8]. Minimizing $\|\mathbf{D}\mathbf{b}\|_1$ makes estimated background and foreground smooth as possible as it can. That is, if a pixel belongs to background (or foreground), then its neighbors also belong to background (or foreground).

3.3. Problem Solution

To obtain the solution to Eq. (3), Eq. (3) is equivalently transformed to the following problem:

$$\min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta (1 - b_i) - \alpha b_i (1 - s_i) \right] + \lambda \|\mathbf{c}\|_1, \quad (4)$$

$$s.t. \mathbf{c} = \mathbf{D}\mathbf{w}, \mathbf{w} = \mathbf{b}. \quad (5)$$

The constrained minimum problem expressed as Eq. (4) and Eq. (5) can be converted to the following unconstrained problem:

$$\begin{aligned} \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}, \mathbf{c}, \mathbf{w}, \mathbf{x}, \mathbf{y}} \sum_{i=1}^N & \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta (1 - b_i) - \alpha b_i (1 - s_i) \right] + \lambda \|\mathbf{c}\|_1 + \frac{\mu}{2} \|\mathbf{w} - \mathbf{b}\|_F^2 \\ & + \mathbf{x}^T (\mathbf{w} - \mathbf{b}) + \frac{\mu}{2} \|\mathbf{c} - \mathbf{D}\mathbf{w}\|_F^2 + \mathbf{y}^T (\mathbf{c} - \mathbf{D}\mathbf{w}). \end{aligned} \quad (6)$$

We adopt an alternative minimization algorithm to seek the optimal solutions of Eq. (6). The steps are as follows.

b – Step. The goal is to seek the optimal \mathbf{b} when \mathbf{U} , \mathbf{v} , \mathbf{c} , \mathbf{w} , \mathbf{x} , and \mathbf{y} are fixed. Computing the derivative of the sum of the terms of Eq. (6) and letting the result be zero yields

$$b_i = \frac{\beta + \mu w_i + x_i - \frac{1}{2} (\mathbf{U}_i \mathbf{v} - o_i)^2 + \alpha (1 - s_i)}{\mu}. \quad (7)$$

The influence of the saliency map s_i on the background b_i is intuitive: b_i decreases with increasing of s_i . Hence, the proposed method tends to let estimated background not contain moving and salient objects whereas let estimated foreground contain moving and salient objects meanwhile.

c – Step. The goal is to seek the optimal \mathbf{c} when \mathbf{b} , \mathbf{U} , \mathbf{v} , \mathbf{w} , \mathbf{x} , and \mathbf{y} are fixed. Omitting irrelevant terms, it is reduced to the following traditional optimization problem:

$$\mathbf{c} = \arg \min_{\mathbf{c}} \lambda \|\mathbf{c}\|_1 + \frac{\mu}{2} \|\mathbf{c} - \mathbf{D}\mathbf{w}\|_F^2 + \mathbf{y}^T(\mathbf{c} - \mathbf{D}\mathbf{w}) \quad (8)$$

$$= \arg \min_{\mathbf{c}} \frac{\lambda}{\mu} \|\mathbf{c}\|_1 + \frac{1}{2} \|\mathbf{c} - \mathbf{m}\|_F^2, \quad (9)$$

where

$$\mathbf{m} = \mathbf{D}\mathbf{w} - \frac{\mathbf{y}}{\mu}. \quad (10)$$

Eq. (9) is standard minimization problem [4] and the solution is given by [8]

$$\mathbf{c} = S_{\frac{\lambda}{\mu}}(\mathbf{D}\mathbf{w} - \frac{\mathbf{y}}{\mu}) \quad (11)$$

with the soft-thresholding (shrinkage) operator $S_{\varepsilon}(x)$ being

$$S_{\varepsilon}(x) = \text{sgn}(x) \max(|x| - \varepsilon, 0) = \begin{cases} x - \varepsilon, & x > \varepsilon \\ x + \varepsilon, & x < -\varepsilon \\ 0 & \text{else} \end{cases} \quad (12)$$

w – Step. The goal is to seek the optimal \mathbf{w} when \mathbf{b} , \mathbf{U} , \mathbf{v} , \mathbf{c} , \mathbf{x} , and \mathbf{y} are fixed. Omitting irrelevant terms, it is reduced to the following minimization problem:

$$\begin{aligned} \mathbf{w} = \arg \min_{\mathbf{w}} & \frac{\mu}{2} \|\mathbf{w} - \mathbf{b}\|_F^2 + \mathbf{x}^T(\mathbf{w} - \mathbf{b}) \\ & + \frac{\mu}{2} \|\mathbf{c} - \mathbf{D}\mathbf{w}\|_F^2 + \mathbf{y}^T(\mathbf{c} - \mathbf{D}\mathbf{w}). \end{aligned} \quad (13)$$

Specifically, the optimal \mathbf{w} is calculated by

$$\mathbf{w} = (\mathbf{I} + \mathbf{D}^T\mathbf{D})^{-1} \left[\mathbf{D}^T(\mathbf{c} + \frac{\mathbf{y}}{\mu}) + \mathbf{b} - \frac{\mathbf{x}}{\mu} \right]. \quad (14)$$

x, y – Step. The goal is to seek the optimal \mathbf{x} and \mathbf{y} when \mathbf{b} , \mathbf{U} , \mathbf{v} , \mathbf{c} , and \mathbf{w} are fixed. Computing the derivative of the sum of the terms of Eq. (6) w.r.t. \mathbf{x} and \mathbf{y} and then letting the result be zero yields the following updating rule:

$$\mathbf{x} \leftarrow \mathbf{x} + \mu(\mathbf{w} - \mathbf{b}), \quad (15)$$

$$\mathbf{y} \leftarrow \mathbf{y} + \mu(\mathbf{c} - \mathbf{D}\mathbf{w}). \quad (16)$$

It is noted that the coefficient μ is updated by

$$\mu \leftarrow a\mu, \quad (17)$$

where a is a parameter and its empirical value is 1.25.

U – Step. The goal is to seek the optimal \mathbf{U} when \mathbf{b} , \mathbf{v} , \mathbf{c} , \mathbf{w} , \mathbf{x} , and \mathbf{y} are fixed. The problem of minimizing Eq. (6) with respect to \mathbf{U} becomes

$$\mathbf{U} = \arg \min_{\mathbf{U}} \sum_i \frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2, \text{ s.t. } \mathbf{U}\mathbf{U}^T = \mathbf{I}, \quad (18)$$

where \mathbf{I} is the identity matrix. It is known that orthogonal matrices representing linear subspaces of the Euclidean space can be represented as points on the Grassmann manifolds [17]. So subspace estimation can be equivalently formulated into an optimization problem on Grassmann manifolds [17]. Defining

$$L_f \triangleq \frac{1}{2} \mathbf{b}(\mathbf{U}\mathbf{V} - \mathbf{o})\mathbf{v}^T \mathbf{U} = \sum_i \frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2, \quad (19)$$

the optimization can be performed by using the gradient $\partial L_f / \partial \mathbf{U}$ on the Euclidean space and the gradient ∇L_f of the Grassmannian [5]. The gradients are given by

$$\frac{\partial L_f}{\partial \mathbf{U}} = \mathbf{b}(\mathbf{U}\mathbf{V} - \mathbf{o})\mathbf{v}^T, \quad (20)$$

and

$$\begin{aligned} \nabla L_f &= (\mathbf{I} - \mathbf{U}\mathbf{U}^T) \frac{\partial L_f}{\partial \mathbf{U}} \\ &= (\mathbf{I} - \mathbf{U}\mathbf{U}^T) \mathbf{b}(\mathbf{U}\mathbf{V} - \mathbf{o})\mathbf{v}^T, \\ &= (\mathbf{I} - \mathbf{U}\mathbf{U}^T) \mathbf{R} \mathbf{v}^T \end{aligned} \quad (21)$$

where the residual vector \mathbf{R} is defined as

$$\mathbf{R} \triangleq \mathbf{b}(\mathbf{U}\mathbf{v} - \mathbf{o}). \quad (22)$$

The solution on the Grassmannian manifolds is [11], [30].

$$\begin{aligned} \mathbf{U} \leftarrow & \mathbf{U} + (\cos(\sigma\eta) - 1) \mathbf{U} \frac{\mathbf{v}}{\|\mathbf{v}\|} \frac{\mathbf{v}^T}{\|\mathbf{v}\|} \\ & - \sin(\sigma\eta) \frac{\mathbf{R}}{\|\mathbf{R}\|} \frac{\mathbf{v}}{\|\mathbf{v}\|}. \end{aligned} \quad (23)$$

v – Step. The low-dimensional representation \mathbf{v} of \mathbf{o} can be simply calculated by

$$\mathbf{v} = \mathbf{U}^T \mathbf{o}. \quad (24)$$

Algorithm 1 summarizes the above steps.

4. Experimental results

We describe intermediate results followed by comparison with state-of-the-art methods. In our experiments, the saliency maps are obtained by the method developed in [36].

Algorithm 1 The proposed method of moving object detection.

Input:

A sequence of frames (images) and the current image is \mathbf{o} . Each image has N pixels.

Output:

Foreground-indicator vector \mathbf{f} corresponding to the current image \mathbf{o} .

1: **Initialization**

2: Initialize parameters $\alpha, \beta, \mu, \lambda$.

3: Initialize $\mathbf{U}, \mathbf{v}, \mathbf{c}, \mathbf{w}, \mathbf{x}$, and \mathbf{y} .

4: Applying some salient object diction algorithm on \mathbf{o} and get the corresponding saliency map \mathbf{s} .

5: Iterating the following steps several loops

6: Begin Iteration:

7: **b - Step:** $b_i = \frac{\beta + \mu w_i + x_i - \frac{1}{2}(\mathbf{U}_i \mathbf{v} - o_i)^2 + \alpha(1 - s_i)}{\mu}$

8: **c - Step:** $\mathbf{c} = S_{\frac{\lambda}{\mu}}(\mathbf{D}\mathbf{w} - \frac{\mathbf{y}}{\mu})$

9: **w - Step:** $\mathbf{w} = (\mathbf{I} + \mathbf{D}^T \mathbf{D})^{-1} [\mathbf{D}^T (\mathbf{c} + \frac{\mathbf{y}}{\mu}) + \mathbf{b} - \frac{\mathbf{x}}{\mu}]$

10: **x, y - Step:** Assign a small number to μ . Update \mathbf{x} and \mathbf{y} by running the following formulas several loops:
 $\mathbf{x} \leftarrow \mathbf{x} + \mu(\mathbf{w} - \mathbf{b}), \mathbf{y} \leftarrow \mathbf{y} + \mu(\mathbf{c} - \mathbf{D}\mathbf{w}), \mu \leftarrow 1.25\mu$.

11: **U - Step:** Assign a small number to η . Update \mathbf{U} by running the following formulas several loops:
 $\mathbf{R} = \mathbf{b}(\mathbf{U}\mathbf{v} - \mathbf{o}), \mathbf{U} \leftarrow \mathbf{U} + (\cos(\sigma\eta) - 1)\mathbf{U} \frac{\mathbf{v}}{\|\mathbf{v}\|} \frac{\mathbf{v}^T}{\|\mathbf{v}\|} - \sin(\sigma\eta) \frac{\mathbf{R}}{\|\mathbf{R}\|} \frac{\mathbf{v}}{\|\mathbf{v}\|}$.

12: End Iteration

13: Compute foreground-indicator vector \mathbf{f} is obtained by binarizing background vector: $f_i = \begin{cases} 0 & \text{if } b_i \geq t, \\ 1 & \text{if } b_i < t. \end{cases}$

4.1. Intermediate results

We give intermediate results to show the role of the saliency map term $-b_i(1 - s_i)$ and the connectivity term $\|\mathbf{D}\mathbf{b}\|_1$.

For notation simplicity, in Table 1 we list the objective functions of three methods: Baseline, Add Connectivity, and Add Saliency Map. The objective function of the proposed method is

$$L = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i) \right] + \lambda \|\mathbf{D}\mathbf{b}\|_1. \quad (25)$$

The baseline is the method whose objective function L_b (Eq. (26)) consists of the first two terms of L (Eq. (25)):

$$L_b = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) \right]. \quad (26)$$

In addition to the reconstruction term, the baseline method merely makes use of the sparsity term $\beta(1 - b_i)$.

Table 1: Method used for intermediate results.

Method	Objective Function
Baseline	$L_b = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) \right]$
Add Connectivity	$L_c = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) \right] + \lambda \ \mathbf{D}\mathbf{b}\ _1$
Add Saliency Map	$L = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) - \alpha b_i (1 - s_i) \right] + \lambda \ \mathbf{D}\mathbf{b}\ _1$

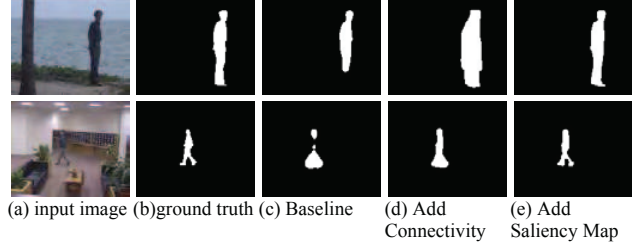


Figure 2: The influence of adding connectivity and saliency map to the objective function.

Compared to L_b (Eq. (26)), the objective function L_c (Eq. (27)) of Add Connectivity has additional connectivity term $\lambda \|\mathbf{D}\mathbf{b}\|_1$:

$$L_c = \min_{\mathbf{b}, \mathbf{U}, \mathbf{v}} \sum_{i=1}^N \left[\frac{1}{2} b_i (\mathbf{U}_i \mathbf{v} - o_i)^2 + \beta(1 - b_i) \right] + \lambda \|\mathbf{D}\mathbf{b}\|_1. \quad (27)$$

The objective function of Add Saliency Map is the same as L (Eq. (25)). That is, Add Saliency Map is the final form of our method where sparsity, low-rank, connectivity, and saliency map are taken into account.

Several frames of the Perception Test Image Sequences [15] are used for analyzing the intermediate results. Some examples are shown in Fig. 2 and Fig. 3. Fig. 2 (a) shows two input frames with water-surface background for the top one and indoor environment for the bottom one. The ground truth of the moving object is given in Fig. 2 (b). Fig. 2 (c) is the detected results of the baseline from which one can see that the detected object is smaller than the ground truth. The top of Fig. 2 (c) shows that the feet and some portions of the shanks are missed by the baseline method. The bottom of Fig. 2 (c) shows that the middle of the person is missed by the baseline method.

As can be seen from Fig. 2 (d), with the help of connectivity term, the Add Connectivity is able to detect the missed parts (feet and legs in the top of Fig. 2 (d) and the middle part of the person in the bottom of Fig. 2 (d)) of the persons. But one can also there are many false alarms in Fig. 2 (d). False alarms are the by-product of Add Connectivity. Fig. 2 (e) is the result of Add Saliency Map. Obviously, introducing the saliency map successfully discards the false alarms existing in Fig. 2 (d).

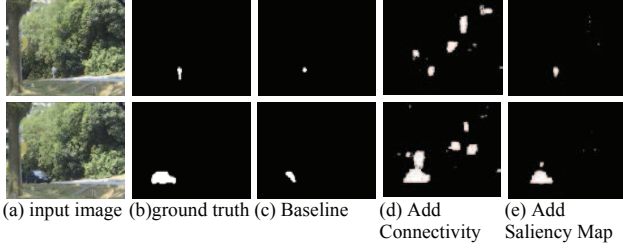


Figure 3: The influence of adding connectivity and saliency map to the objective function.

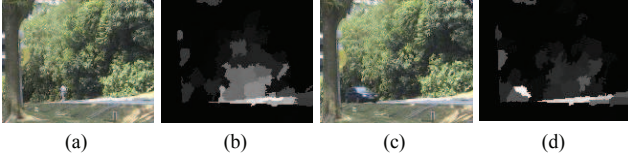


Figure 4: (b) and (c) are the saliency maps of (a) and (c), respectively.

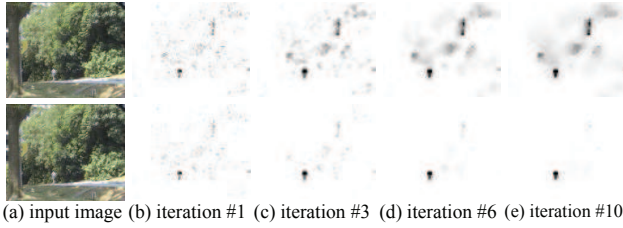


Figure 5: Input image and the background vector obtained in different iteration of the Add Connectivity (Top) and Add Saliency Map (bottom).

The results given in Fig. 3 (e) demonstrate that adding saliency map into the objective function is capable of suppressing many false alarms when the size of moving objects (a person in the top of Fig. 3 (a) and a car in the bottom of Fig. 3 (a)) is small whereas the background is large, complex and dynamic. Fig. 3 (d) shows that adding connectivity into the objective function not only enlarges the objects detected by the baseline but also incorrectly classifies moving leaves and shadows as semantic objects. Adding saliency map (Fig. 3 (e)) plays a role of overcoming the drawback of adding connectivity.

The saliency maps of the top and bottom of Fig. 3 (a) are shown in Fig. 4 (b) and Fig. 4 (d), respectively. Though the saliency maps are not ideal, they provide useful clue for the proposed method (i.e., Add Saliency Map).

The proposed Add Saliency Map algorithm (see Algorithm 1) and Add Connectivity algorithm update the background vector \mathbf{b} iteratively. Fig. 5 and Fig. 6 show how the background vector \mathbf{b} varies with iterations. Fig. 5 (a) shows the input image identical to the top of Fig. 3 (a). The top and bottom of Fig. 5 corresponds to the iteration results of Add

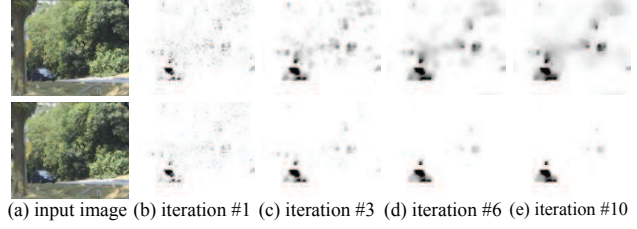


Figure 6: Input image and the background vector \mathbf{b} obtained in different iteration of the Add Connectivity (Top) and Add Saliency Map (bottom).

Connectivity algorithm and Add Saliency Map algorithm, respectively. One can see from the top of Fig. 5 that the background vector obtained by Add Connectivity contains more regions of waving leaves as iteration proceeds. But one can see from the bottom of Fig. 5 that the background vector obtained by Add Saliency Map excludes more regions of waving leaves as iteration proceeds and hence the foreground vector focuses on the true meaningful moving person.

Similar to Fig. 5, the bottom of Fig. 6 also demonstrates that adding the saliency map into the objective function makes the estimated background vector iteratively excludes the influence of moving leaves.

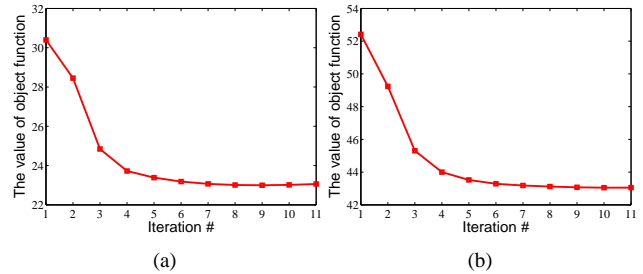


Figure 7: Convergence of the Add Saliency Map. (a) For the input image shown in Fig. 5(a). (b) For the input image shown in Fig. 6(a).

Fig. 7 shows that the convergence of the proposed algorithm. Generally, the value of the objective function L decreases drastically at the first five iterations and becomes stable after iteration # 8.

4.2. Comparison with the State-of-the-art methods

We call the proposed method Moving Object Detection using Saliency Map with abbreviation MODSM.

The Perception Test Images Sequences [15] are also used for comparison with the state-of-the-art methods. The dataset consists of 9 videos captured in a variety of indoor and outdoor environments, including offices, campuses, sidewalks, and other private and public sites.

The weather conditions when collecting the data cover

sunny, cloudy, and rainy weather. The videos with static background are named Bootstrap (BS), Shopping Mall (SM), and Hall (Hal). The videos with dynamic background are called Fountain (Fou), Escalator (Esc), Water Surface (WS), Curtain (Cur), and Campus (Cam). The Lobby (Lob) video is captured when there are drastic variations in illumination. The sizes (widths and heights) of the frames includes [160, 130], [160, 128], [176, 144], [160, 120], [160, 128], and [320, 256].

We compare the proposed MODSM algorithm with PCP (Principal Component Pursuit) [3], DP-GMM (Dirichlet Process Gaussian Mixture Models) [9], GMM [24], GRASTA (Grassmannian Robust Adaptive Subspace Tracking Algorithm) [11], DECOLOR (DEtecting COntiguous Outliers in the LOw-rank Representation) [35], SOBS [16] and RFDSA [8]. PCP, GRASTA, and RFDSA are the state-of-the-art subspace based algorithms. DP-GMM is the state-of-the-art density based algorithm and DECOLOR is the state-of-the-art low-rank based algorithm. DP-GMM and GRASTA, are incremental algorithms whereas PCP, DECOLOR, and RFDSA are batch algorithms. We run the source codes provided by the authors of four methods on the dataset to get the experimental results. Note that GRASTA randomly samples a fraction of pixels in an image for subspace modeling and object detecting. Its detection accuracy increases with the fraction. To reduce randomness and get its best accuracy, 100% pixels are used in our experiments.

The parameters (see Eq. (6)) of the MODSM method are given in Table 2 where m is the number of columns (basis vectors) of the matrix \mathbf{U} . In Table 2, $\hat{\sigma}^2$ is given by

$$\hat{\sigma}^2 = \frac{1}{2|\Omega|} \sum_{\mathbf{o} \in \Omega} \|\mathbf{U}\mathbf{v} - \mathbf{o}\|_2^2, \quad (28)$$

where Ω and $|\Omega|$ are the set and the number of training images, respectively. s_m is the ratio of the number of pixels whose saliency are larger than the mean of saliency maps of training images:

$$s_m = \frac{\sum_{\mathbf{s}} \sum_{i=1}^N I(s_i - s_M)}{N|\Omega|}, \quad (29)$$

with

$$s_M = \frac{\sum_{\mathbf{s}} \sum_{i=1}^N s_i}{N|\Omega|}, \quad (30)$$

and

$$I(x) = \begin{cases} 1 & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (31)$$

Note that the $\lceil x \rceil$ in Table 2 stands for the floor function of x .

Table 2 gives a general rule for parameter setting. But the detection performance can be significantly improved if video-specific parameters are utilized.

The F_1 -score, the harmonic mean of precision and recall, is used for objective evaluation:

$$F_1 = 2 \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}. \quad (32)$$

The results of the different methods are given in Table 3. Among the nine videos, the proposed MODSM, RFDSA, and DECOLOR get the best performance on five (i.e., WS, Cur, Hal, Esc, and BS), two (i.e., SM and Lob), and two (Fou and Cam) different videos, respectively. The average F_1 -score of the proposed MODSM is the largest. But our method does not work well for the Lobby (i.e., Lob) video. The main reason is that the performance of the method [36] of creating saliency map on the Lobby video degraded significantly. If the Lobby video is excluded, the average F_1 -score of MODSM grows from 0.7711 to 0.7955 whereas that of RFDSA decreases from 0.7489 to 0.7421. It is expected that the performance of MODSM increases with the performance of saliency map.

Table 3 also shows that if proper prior information (i.e., connectivity, saliency map, sparsity) is employed then the incremental algorithm MODSM can outperform the batch algorithms DECOLOR and RFDSA.

The ROC curves of the MODSM and RFDSA on the Water Surface, Escalator, and Fountain, and Campus videos are shown in Fig. 8 where the superiority of the MODSM can be observed. Take the Fountain video as an example. The true positive rates (i.e., recall) of MODSM and RFDSA are respectively 0.99 and 0.935 when the false positive rate is 0.05. Note that the DECOLOR method cannot generate the ROC curves because of their binary values of the estimated foreground and background.

Several specific results of MODSM, RFDSA, and DECOLOR are visualized in Fig. 9, Fig. 10, and Fig. 11 where (a), (b), (c), (d), and (e) are the current input frame,

Table 2: Parameters of the MODSM method.

m	β	λ	α	μ
5	$\beta = \max(\frac{1}{2}\beta, 4.5\hat{\sigma}^2)$	5β	$\min\left(\frac{s_m}{s_m - s_M}, \hat{\sigma}s_m, 6.5\beta\right)$	0.1

Table 3: F1-scores of different methods.

Method	WS	Cur	Fou	Hal	SM	Lob	Esc	BS	Cam	mean
GMM	.7948	.7580	.6854	.3335	.5363	.6519	.1388	.3838	.0757	.4842
SOBS	.8247	.8178	.6554	.5943	.6677	.6489	.5770	.6019	.6960	.6760
DP-GMM	.9090	.8203	.7049	.5484	.6522	.5794	.5055	.6024	.7567	.6754
PCP	.4137	.6193	.5679	.5917	.7234	.6989	.6728	.6582	.3406	.5874
DECOLOR	.8866	.8255	.8598	.6424	.6525	.6149	.6994	.5869	.8096	.7308
GRASTA	.7310	.6591	.3786	.5817	.7142	.5550	.4697	.6146	.2504	.5505
RFDSA	.8796	.8976	.7544	.6673	.7407	.8029	.6353	.6841	.6779	.7489
MODSM	.9404	.9098	.8205	.6859	.7362	.5762	.7553	.7280	.7876	.7711

ground truth of the moving objects, the detected results of MODSM, RFDSA, and DECOLOR, respectively.

Fig. 9 (a) is a frame of the Curtain video. Fig. 9 (d) shows that RFDSA incorrectly regards the variation caused by motion of the curtain as moving objects and RFDSA results in incomplete neck of the person. Fig. 9 (e) shows that DECOLOR gives rise to even more false alarms. Investigating Figs. 9 (c) and (b), one can find the result of MODSM is very close to the ground truth.

Fig. 10 (a) is a frame of the Campus video. Fig. 10 (d) shows that RFDSA incorrectly classifies many waving leafs as meaningful moving objects. Fig. 10 (e) tells that DECOLOR cannot detect the left small person and the head of the right large person is also mistakenly classified as back-ground. Fig. 10 (c) shows that the proposed method is powerful for classifying the waving leafs as background and detecting both of the persons.

Fig. 11 (a) is a frame of the Escalator video. Fig. 11 (d) shows that RFDSA classifies moving escalator as semantically meaningful moving objects. Because of using the information of saliency map, the proposed MODSM (Fig. 11 (c)) avoids the errors of RFDSA. Fig. 11 (e) shows that DECOLOR has almost not missed alarms but has many false alarms. The result (Fig. 11 (c)) of MODSM is the best among the three methods.

Fig. 12 (a) is a frame of the Shopping Mall video. It can be seen that MODSM is comparable and even slightly better than RFDSA and DECOLOR.

As can be seen from Table 3, the proposed method MODSM results unsatisfying results on the Lobby video.

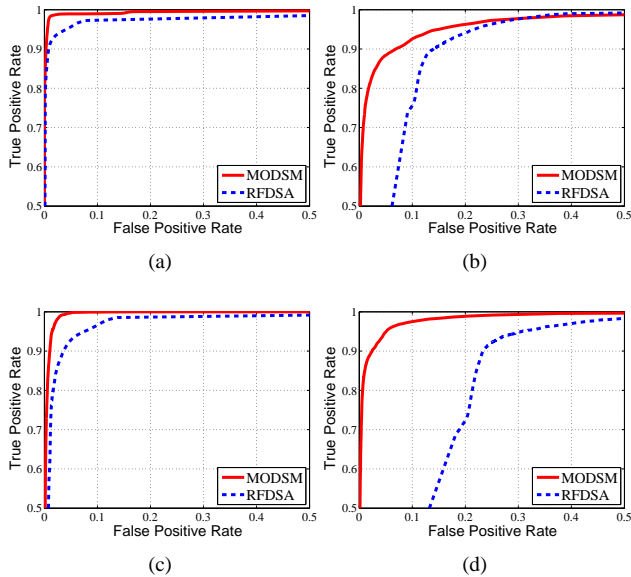


Figure 8: ROC curves on the Water Surface, Escalator, Fountain, and Campus video.

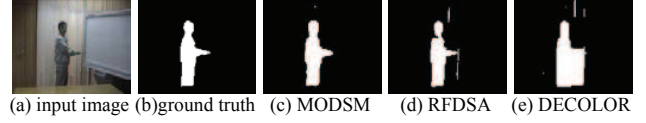


Figure 9: Detected objects for a frame of the Curtain video.

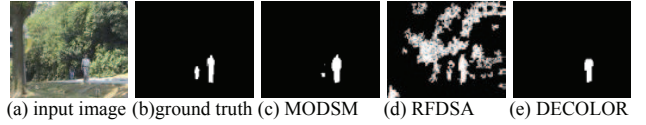


Figure 10: Detected objects for a frame of the Campus video.



Figure 11: Detected objects for a frame of the Escalator video.



Figure 12: Detected objects for a frame of the Shopping Mall video.



Figure 13: Detected result for a frame of the Lobby video.

Fig. 13 attempts to explain the reason. On the one hand, switching from light on (Fig. 13 (a)) to light off (Fig. 13 (b)) gives rise to large variation which is difficult for the basis vectors U to capture. On the other hand, the saliency map is not satisfying on the regions of the moving object (person). In this case, introducing the bad saliency map (Fig. 13 (c)) has a negative influence on the task of moving object detection. The research progress of salient object detection is helpful for improving the performance of the propose method.

5. Conclusion and future work

In this paper, we have presented a moving object detection method. The method makes use of saliency map by incorporating it into a unified objective function where the properties of sparsity, low-rank, connectivity, and saliency are integrated. The manner of using saliency map yields smaller number of false alarms and missed alarms. Our future work will apply the idea of using saliency map to other

state-of-the-art incremental and batch methods of moving object detection. Moreover, we will investigate other state-of-the-art methods of generating saliency map.

References

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, SLIC superpixels compared to state-of-the-art superpixel methods, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
- [2] L. Balzano, R. Nowak, and B. Recht, Online identification and tracking of subspaces from highly incomplete information, *Proc. Allerton Conference on Communication*, 2010.
- [3] E. Candes, X. Li, Y. Ma, and J. Wright Robust principal component analysis? *Journal of the ACM*, vol. 58, no. 3, pp. 1-37, 2011.
- [4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1-22, 2011.
- [5] A. Edelman, T. A. Arias, and S. T. Smith, The geometry of algorithms with orthogonality constraints, *SIAM Journal on Matrix Analysis and Applications*, vol. 20, no. 2, pp. 303-353, 1998.
- [6] P. Dollár, R. Appel, S. Belongie, and P. Perona, Fast feature pyramids for object detection, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532-1545, 2014.
- [7] P. Favaro, R. Vidal, and A. Ravichandran, A closed form solution to robust subspace estimation and clustering, *CVPR*, 2011.
- [8] X. Guo, X. Wang, L. Yang, X. Cao, and Yi Ma, Robust foreground detection using smoothness and arbitrariness constraints, *Proc. European Conference on Computer Vision*, 2014.
- [9] Tom S.F. Haines and T. Xiang, Background subtraction with Dirichlet process mixture models, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 670-683, 2014.
- [10] H. Hao, S. Li, C. Zhu, H. Chang, J. Zhang, Moving object detection in aerial video based on spatiotemporal saliency, *Chinese Journal of Aeronautics*, vol. 26, no. 5, pp. 1211-1217, 2013.
- [11] J. He, L. Balzano, and A. Szlam, Incremental gradient on the Grassmannian for online foreground and background separation in subsampled video, *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2012.
- [12] I. Haritaoglu, D. Harwood, and L. S. Davis, W4: real-time surveillance of people and their activities, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 809-830, 2000.
- [13] M. Isard and A. Blake, CONDENSATION - conditional density propagation for visual tracking, *Int. J. Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.
- [14] H. Li, H. Lu, Z. Lin, X. Shen, and B. Price, Inner and inter label propagation: salient object detection in the wild,” *IEEE Trans. Image Processing*, vol. 24, no. 10, pp. 3176-3186, 2015.
- [15] L. Li, W. Huang, I. Y. H. Gu, and Q. Tian, Statistical modeling of complex backgrounds for foreground object detection, *IEEE Trans. Image Processing*, vol. no. 11, pp. 1459-1472, 2004.
- [16] L. Maddalena and A. Petrosino, A self-organizing approach to background subtraction for visual surveillance applications, *IEEE Trans. Image Processing*, vol. 17, no. 7, pp. 1168-1177, 2008.
- [17] S. Mittal and P. Meer, Conjugate gradient on Grassmann manifolds for robust subspace estimation, *Image and Vision Computing*, vol. 30, nos. 6-7, pp. 417-427, 2012.
- [18] A. Neri, S. Colonnese, G. Russo, and P. Talone, Automatic moving object and background separation, *Signal Processing*, vol. 66, pp. 219-232, Apr. 1998.
- [19] Y. Pang, K. Zhang, Y. Yuan, and K. Wang, Distributed object detection with linear SVMs, *IEEE Trans. Cybernetics*, vol. 44, no. 11, pp. 2122-2133, 2014.
- [20] Y. Pang, S. Wang, and Y. Yuan, Learning regularized LDA by clustering, *IEEE Trans. Neural Networks and Learning Systems*, vol. 25, no. 12, pp. 2191-2201, 2014.
- [21] Y. Pang, X. Li, J. Pan, and X. Li, Incrementally detecting moving objects in video with sparsity and connectivity, *Cognitive Computation*, 2015.
- [22] C. Qiu and N. Vaswani. Reprocs, ”Missing link between recursive robust pca and recursive sparse recovery in large but correlated noise, *arXiv*, 1106.3286, 2011.
- [23] M. Shakeri and H. Zhang, COROLA: a sequential solution to moving object detection using low-rank approximation, *CoRR*, abs/1505.03566, 2015.
- [24] C. Stauffer and W. Grimson, Adaptive background mixture models for real-time tracking, *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 1999.
- [25] F. Torre and M. Black. A framework for robust subspace learning, *IJCV*, 54(1):117-142, 2003.
- [26] R. Vidal, Y. Ma, and S. Sastry, Generalized principal component analysis (GPCA), *IEEE Trans. Pattern Analysis and Machine Intelligence*, Neurocomputing, vol. 27, no. 12, pp. 1945-1959, 2005.
- [27] W. Wang, J. Shen, X. Li, and F. Porikli, Robust video object cosegmentation, *IEEE Trans. Image Processing*, vol. 24, no. 10, pp. 3137-3148, 2015.
- [28] K. Wang, L. Xu, Y. Fang, and J. Li, One-against-all frame differences based hand detection for human and mobile interaction, *Neurocomputing*, vol. 120, pp. 185-191, 2013.
- [29] Y. Xia, R. Hu, Z. Wang, and T. Lu, Moving foreground detection based on spatio-temporal saliency, *International Journal of Computer Science Issues*, vol. 10, no. 1, pp. 79-84, 2013.
- [30] J. Xu, V. K. Ithapu, L. Mukherjee, J. M. Rehg, and V. Singh, GOSUS: Grassmannian online subspace updates with structured-sparsity, *Proc. IEEE International Conference on Computer Vision*, 2013.

- [31] B. Xin, Y. Tian, Y. Wang, and W. Gao, Background Subtraction via Generalized Fused Lasso Foreground Modeling, *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2015.
- [32] Y. Yuan, Y. Pang, J. Pan, and X. Li, Scene segmentation based on IPCA for visual surveillance, *Neurocomputing*, vol. 72, nos. 10-12, pp. 2450-2454, 2009.
- [33] S. Zhong, Y. Liu, F. Ren, J. Zhang, T. Re, Video saliency detection via dynamic consistent spatio-temporal attention modelling, *Proc. AAAI Conference on Artificial Intelligence*, 2013.
- [34] X. Zhou, C. Yang, H. Zhao, and W. Yu, Low-rank modeling and its applications in image analysis, *CoRR*, abs/1401.3409, 2014.
- [35] X. Zhou, C. Yang, and W. Yu, Moving object detection by detecting contiguous outliers in the low-rank representation, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 597-610, 2013.
- [36] W. Zhu, S. Liang, Y. Wei, and J. Sun, Saliency optimization from robust background detection, *Proc. IEEE International Conference on Computer Vision and Pattern Recognition*, 2014.